基于序列到序列神经网络模型的古诗自动生成方法 *

黄文明 [1,2]，卫万成 [1]，邓珍荣 [1,2]

(1. 桂林电子科技大学 计算机与信息安全学院，广西 桂林 514000; 2. 广西高校云计算与复杂系统重点实验室，广西 桂林 514000)

摘　要：计算机写诗是实现计算机写作的第一步。目前计算机写诗普遍存在主题不明确、诗的内容与写作意图不一致的问题。为改善这些问题，效仿古人写诗的过程，提出了一种分为两个阶段生成古诗的方法。第一阶段获取写诗大纲，此过程采用 TextRank 算法对用户输入文本提取关键词，并提出一种基于注意力机制的序列到序列神经网络模型用于关键词扩展；第二阶段根据写诗大纲生成每一行诗句，此过程提出一种包含双编码器和注意力机制的序列到序列神经网络模型用于古诗生成。最后通过对实验结果的评估验证了所提方法的有效性。与基准方法相比，所提方法生成的古诗，主题意义更加明确，诗所表现的内容和写作意图更加一致。

关键词：关键词扩展；注意力机制；序列到序列；神经网络模型；古诗生成

中图分类号：TP391　　doi: 10.3969/j.issn.1001-3695.2018.03.0371

Automatic generation of poetry based on sequence-to-sequence neural network model

Huang Wenming[1, 2], Wei Wancheng[1], Deng Zhenrong[1, 2]

(1. *College of Computer & Information Security*, *Guilin University of Electronic Technology*, *Guilin* 541004, *China*; 2. *Guangxi Colleges & Universities Keys Laboratory of Cloud Computing & Complex Systems*, *Guilin* 541004, *China*)

Abstract: Computer poetry generation is our first step towards computer writing. At present, there are many problems in computer poetry writing, such as unclear theme, the content of poetry is inconsistent with the writing intention. For improving these problems, this paper follows the process of writing poem by the ancient Chinese poet and proposes a method for generating Chinese poetry with two stages. The first stage extracts the outline. During this process, this paper use TextRank algorithm to extract keywords from user input text, and propose an attention-based sequence to sequence neural network model for expanding keyword. The second stage generates each line of poem based on the outline of poem. During this process, this paper proposed a sequence to sequence neural network model with dual-encoding and attention mechanism for generating poem. At the end, this paper verified the effectiveness of our approach by evaluation. Compared with baseline approach, the theme of the poem generated by our approach is more explicit, and the contents expressed by the poem are more consistent with the writing intention.

Key words: keywords expansion; attention mechanism; sequence to sequence; neural network model; Chinese poetry generation

0　引言

古诗是中国文化的精粹。古诗一般被用来歌颂英雄人物、美丽的风景、爱情、友谊等。古诗被分为很多类，唐诗、宋词、元曲等，每种古诗都有自己独特的结构、韵律。表 1 中展示了一种中国古代最流行的古诗体裁——唐诗绝句。绝句在结构和韵律上具有严格的限制：每首诗由四行组成，每一行有五个或者七个汉字(五个汉字称为五言绝句，七个汉字称为七言绝句)；每个汉字音调要么是平，要么是仄；诗的第二行和最后一行的最后一个汉字必须同属于一个韵部[1]。正因为绝句在结构和韵律上具有严格的限制，所以好的绝句朗诵起来节奏感很强。

表 1　唐诗绝句

春晓
春眠不觉晓，（平平仄平仄）
处处闻啼**鸟**。（仄仄平平仄）
夜来风雨声，（仄平平仄平）
花落知多**少**。（平仄平平仄）

近几年，古诗自动生成研究得到了学术界的广泛关注。科研者们采用了各种方法研究古诗生成，文献[2～6]采用规则和

收稿日期：2018-03-21；**修回日期**：2018-07-29　**基金项目**：广西高校云计算与复杂系统重点实验室资助项目（yf17106）；广西自然科学基金资助项目（2018GXNSFAA138132）；桂林电子科技大学研究生教育创新计划资助项目（2018YJCX55）

作者简介：黄文明（1963-），男，江苏苏州人，教授，主要研究方向为自然语言处理、图形图像处理；卫万成（1993-），男，江苏连云港人，硕士研究生，主要研究方向为自然语言处理（824396385@qq.com）；邓珍荣，女，研究员，广西桂林，主要研究放向为自然语言处理、图形图像处理.

模板的方式，文献[7～9]采用文本生成算法生成古诗，文献[10]采用自动摘要的方法，文献[11,12]采用统计机器翻译的方法。最近，深度学习方法被广泛的应用于各种自然语言生成任务上，并取得了很大的成效。在古诗生成任务上，文献[13～16]提出了新的思路，将古诗生成看成是一种序列到序列的生成问题，通过用户输入的写作意图生成诗的第一行，然后下面每行根据已生成的诗句顺序地生成。这种生成方法取得了很大的进步，也让古诗生成在效果上上了很大一个台阶，但是生成方法还是存在着一定的问题。采用这种方法用户写作意图仅仅对第一行诗的生成产生了较大的影响，对其他三行诗的生成没有什么影响，这会导致生成出来的诗所表现的内容和写作意图不一致，古诗所表现的主题不明确。

本文效仿了古人作诗的过程，提出了一种分两个阶段生成古诗的方法：第一阶段根据用户输入写作意图得到作诗的大纲；第二阶段根据大纲利用具有双编码器和注意力机制的序列到序列模型顺序地生成诗的每一行。在第一阶段中，用户首先输入写作意图；然后采用 TextRank 算法提取关键词，并提出采用一种基于注意力机制的序列到序列模型对关键词扩展，构建出 N 个相互联系的关键词作为诗的大纲，用于生成具有 N 行诗句的古诗。在第二阶段中，每行诗句对应一个关键词，将关键词和已生成的诗句作为具有双编码器和注意力机制的序列到序列模型的输入，顺序地生成整首诗。本文方法由于从写作意图中扩展出诗的大纲，大纲的关键词之间相互联系，诗又是严格按照大纲来生成，所以生成出的诗，主题明确，内容一致，且紧扣写作意图。

1 相关工作

在自然语言处理中，古诗生成是一项非常具有挑战性的任务。文献[5,6]提出了一种基于语义和语法模板的西班牙诗生成方法。文献[4]采用基于词联想的方法生成俳句。文献[2,3]采用短语搜索的方法生成日本诗。文献[16]采用统计的方法对格律诗进行分析、生成。文献[17]采用了严格的模板方式实现了一个基于语料库生成诗歌的系统。文献[10]认为诗歌生成是个可优化问题，采用基于摘要框架并结合一些规则的方法生成诗歌。文献[7～9]采用了一些生成算法生成古诗，其中统计机器翻译（statistical machine translation，SMT）算法是一种很有效的方法。文献[18]采用一种基于 SMT 的模型来自动生成汉语对联，对联可以被看做是只有两行的诗句，第一行被视为源语言，翻译出第二行。文献[11]对这种方法作了延伸，将 SMT 的模型用来生成绝句，根据前面的行生成后面的行。

最近，深度学习方法在诗生成任务上获得了很大的成功。文献[12]提出了基于循环神经网络（RNN）的绝句生成方法，这种方法首先根据给定的关键词利用 2010 年 Mikolov 等人[19]提出的循环神经网络语言模型（RNNLM）生成诗的第一行，然后后面行根据前面已经生成的所有行顺序的生成，最后组成一首诗。文献[13]采用一个端到端的神经机器翻译模型生成宋词，通过翻译先前行得出下一行。这种方法类似于 SMT，但是在两句话之间的语义相关性更好，在文献[13]中没有考虑第一行诗的生成，所以需要用户输入诗的第一行。文献[15]将这种方法应用在绝句生成上面，并解决第一行诗的生成问题，根据用户输入，然后利用一个单独的神经机器翻译模型（NMT）将其翻译成诗的第一行。文献[20]提出一个新的诗歌生成算法，首先根据输入的关键词生成出相关的韵文，然后根据韵文利用序列到序列模型[22]生成整首诗。

以上古诗生成方法都是根据用户输入文本，然后生成与之对应的古诗。研究发现，采用以上方法生成的古诗存在用户输入受限、诗表达主题不明确、内容与写作意图不一致等问题。为改善这些问题，本文提出了新颖的方法。其贡献如下：首先，针对以上方法仅允许用户输入关键词或者需要用户给出诗的第一行的问题，采用对用户输入文本提取关键词的方法，使用户输入不受限制，用户输入可以是一个词，一个句子，甚至一段话；其次，针对诗表达主题不明确的问题，首次提出采用一种基于注意力机制的序列到序列模型对大纲关键词扩展，模型中引入的注意力机制和双向 LSTM，使所扩展的关键词之间联系大大增强，且一定程度上能够体现出古诗的主题情感；最后，针对诗表达内容与写作意图不一致的问题，提出采用一种具有双编码器和注意力机制的序列到序列模型生成古诗，将大纲关键词和已生成的诗句作为模型的输入，输出每一行诗句，每行诗句严格按照大纲来生成，使生成的古诗内容与写作意图一致。

2 方法实现

2.1 概述

该文将古诗生成分为两个阶段，第一阶段根据用户输入的意图构建作诗大纲，第二阶段根据作诗大纲生成整首古诗。图 1 展示了古诗生成的整个流程。假设一首古诗由 N 行诗句组成，L_i 代表第 i 行诗。第一阶段中，根据用户输入的意图，构建出 N 个关键词 $(K_1,K_2,K_3,...,K_N)$，关键词就是作诗大纲。K_i 表示第 i 个关键词，在生成阶段作为第 i 行诗句的子标题。第二阶段中，将 K_i 和 $L_{1:i-1}$ 作为输入，生成 L_i，其中 $L_{1:i-1}$ 为已生成的所有行诗。每行诗根据作诗大纲给的一个子标题和之前生成的所有行诗句进行生成，如此，顺序地生成整首诗。

2.2 大纲构建

假如要生成的一首诗有 N 行，那么需要构建 N 个相互之间具有联系的关键词来作为大纲，一个关键词作为一行诗句的子标题。首先，根据用户的输入提取关键词。假设用户的输入为 A，A 有长有短，从 A 中本文提取出来的关键词数要小于等于 N。如果 A 很长，那么提取其中最重要的 N 个关键词作为作诗大纲。如果 A 较短，从 A 中提取出的关键词小于 N 个，那么需要将关键词个数扩展成 N 个。

2.2.1 关键词提取

首先，从用户输入文本中提取关键词。本文采用 TextRank 算法[22]评估词在一句话或者一段话中的重要程度。TextRank 是

由 PageRank 算法[23]演化而来，是一种基于图排序的算法。在 TextRank 算法中，由节点及节点间的连接关系构成一个无向的网络图，节点之间的权重根据两个词的总计共现次数来设定。根据 TextRank 最终得分进行排序，得出用户输入文本中最关键的 M 个词 $(M \leqslant N)$。一开始，给 $S(V_i)$ 一个初始化值，然后根据式(1)进行迭代，计算得分，直到收敛。

$$S\left(V_i\right)=\left(1-d\right)+d\sum_{V_j\in E\left(V_i\right)}\frac{\omega_{ji}}{\sum_{V_k\in E\left(V_j\right)}\omega_{jk}}S\left(V_j\right) \tag{1}$$

其中：ω_{ji} 是节点 V_j 和 V_i 连接边的权重；$E(V_i)$ 表示与 V_i 连接的节点的集合；d 表示阻尼因子，通常设为 0.8[27]；$S(V_i)$ 为节点 V_i 的 TextRank 得分，初始分被设为 1.0。

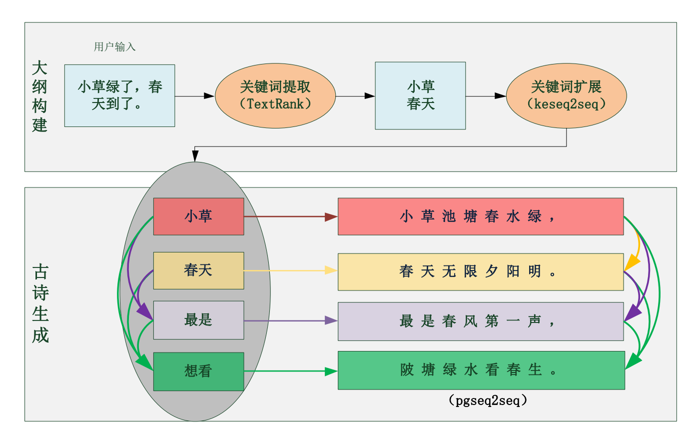


图 1　古诗生成框图

2.2.2 关键词扩展

一般情况下，从用户输入 A 中提取的关键词 M 都会小于 N，此时需要对关键词进行扩展。关键词作为作诗大纲，导向诗的主题及意义，诗生成的好坏，关键词起到了重要的作用。从用户输入文本中提取的关键词，受限于用户输入，而关键词的扩展不受任何限制，因此对关键词扩展作深入研究很有必要。本文在关键词扩展方面进行了深入研究，力求关键词的扩展贴近诗人的联想。本文尝试了以下三种方法对关键词作扩展：基于神经网络语言模型扩展方法；基于 word2vec 词向量模型的扩展方法；基于注意力机制的序列到序列模型扩展方法。

1）基于神经网络语言模型扩展方法

本文将循环神经网络语言模型（recurrent neural network language model，RNNLM）[19]中循环神经网络（recurrent neural network，RNN），用门控循环单元（gated recurrent unit，GRU）[24]代替。众所周知，GRU 相对于 RNN 能够更好地学习到时序数据之间的长期依赖。本文采用模型根据已有的关键词去扩展其他的关键词，扩展公式为 $K_i=\arg\max_K P\left(K\mid K_{1:i-1}\right)$。其中：$K_i$ 是第 i 个关键词；$K_{1:i-1}$ 是 K_i 之前的所有关键词序列。模型的训练数据是从训练古诗中提取的关键词序列。使用 TextRank 算法从每行诗句中提取一个得分最高的关键词作为诗句的标题。如果一首诗由 N 行诗句组成，就提取 N 个关键词，组成一个关键词序列。从所有收集的古诗中提取所有关键词序列，组成一个训练语料库，用来训练模型。

2）基于 word2vec 词向量模型的扩展方法

word2vec 模型是 Google 在 2013 年开源地将词表征为实数值向量的一种高效的算法模型。通过语料库的训练，词可以用 T 维向量空间表示，而向量空间上的相似度可以用来表示文本语义上的相似。本文用收集到的所有古诗作为语料库，训练 word2vec 词向量模型，词向量维度 T 取 100。K_i 表示第 i 个关键词，当 $i<N$ 时，使用 word2vec 词向量模型对关键词进行扩展，寻找与 K_i 在向量空间上相似的词，取其中相似度最高的词作为 K_i 的扩展词，最终扩展成 N 个关键词。

3）基于注意力机制的序列到序列模型扩展方法

本文将词扩展看成是一个序列到序列的问题，并首次将注意力机制和双向长短时记忆网络（bidirectional long short-term memory net，BiLSTM）应用于关键词扩展模型。模型输入序列是从写作意图中提取出的和当前模型已生成的所有关键词，输出序列是预测出的关键词。在序列到序列模型中，模型输入序列被转换成可以代表其语义的隐层状态的过程叫编码，根据隐层状态规律地生成目标关键词序列的过程叫解码。本文借鉴 Bahdanau 等人[24]的基于注意力机制的序列到序列生成模型，提出基于注意力机制的序列到序列的关键词扩展模型（keseq2seq），模型结构如图 2 所示。

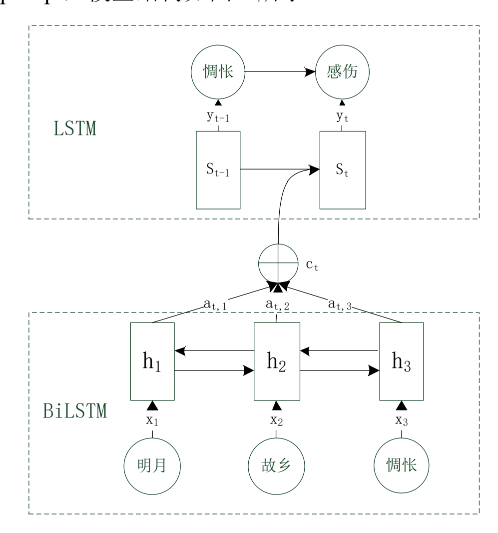


图 2　基于注意力机制的序列到序列关键词扩展模型

一般序列到序列模型中，编码器和解码器采用两个 RNN。因为 RNN 不能很好地学习历史信息，而长短时记忆网络（long

short-term memory net，LSTM）很好地弥补了这个问题。在 keseq2seq 中，本文引入 BiLSTM 作为编码器，并融合注意力机制，LSTM 作为解码器。其中，BiLSTM 不仅能很好地学习历史信息，还能学习未来信息。引入的注意力机制在每个生成时刻，能够更加关注与之密切相关的输入词。本文提出的 keseq2seq 模型实现过程如下：编码器将输入序列 $(X_1,X_2,\dots)$编码成隐层状态$(h_1,h_2,\dots)$，其中，X_i 为第 i 个关键词编码向量；解码器用隐层状态$(h_1,h_2,\dots)$生成输出序列$(y_1,y_2,\dots)$。每个生成时刻，向量 y_t 根据上一时刻的向量 y_{t-1} 和当前状态 S_t 以及当前的文本语义向量 c_t 进行生成，其中 c_t 由编码器的隐藏层状态$(h_1,h_2,\dots)$乘上注意力权重 $a_{t,i}$ 得来。注意力机制中，每个隐藏状态 h_i 对预测 y_t 的贡献程度由注意力权重 $a_{t,i}$ 控制，根据 S_{t-1} 和 h_i 相关度得出 $a_{t,i}$，通过权重 $a_{t,i}$ 的控制，解码器将会更加注意与当前生成密切相关的输入部分。在 keseq2seq 词扩展模型中，引入 BiLSTM 和注意力机制可以大大增强了词与词之间的联系。

以上方法 1）和 2），在一般关键词扩展应用中常被采用，方法 3）的关键词扩展方法由本文首次提出。在论文实验结果中，本文只采用了方法 3)，因为在古诗生成任务上，很明显地发现第三种方法优于前两种方法。基于注意力机制的序列到序列词扩展模型能够很好地学习一个词与另一个词之间的联系。本文采用从训练古诗中提取关键词，组成语料库，用于训练 keseq2seq 模型，让模型学习古诗中前面诗句的关键词与下面诗句的关键词之间的联系。如此训练，模型可以根据输入词，输出与之紧密联系的词，这个过程像是诗人在联想。

2.3 古诗生成模型

论文同样将古诗生成过程看成是一个序列生成另一个序列，与 keseq2seq 不同的是输入由两个序列组成：规定的关键词和所有已生成的诗句。本文对基于注意力机制的序列到序列模型[25]进行改进，让模型能够支持多序列输入。图 3 展示了修改后的具有双编码器和注意力机制的序列到序列古诗生成模型（pgseq2seq）结构。

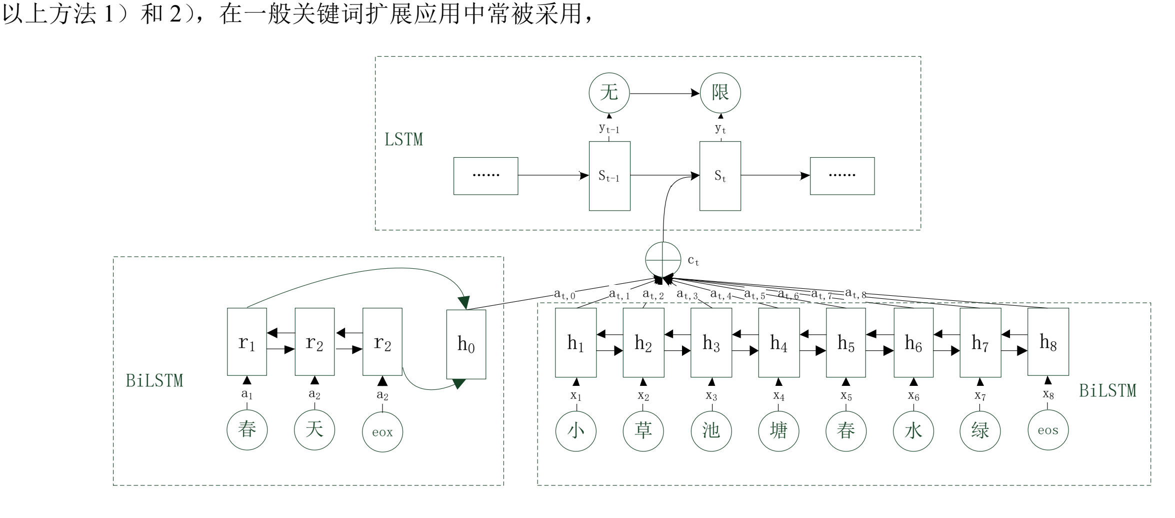


图 3 具有双编码器和注意力机制的序列到序列古诗生成模型

假如关键词 K 有 T_k 个字符，$K=\{a_1,a_2,a_3,\dots,a_{T_k}\}$，已生成的文本 X 有 T_X 个字符，$X=\{x_1,x_2,x_3,\dots,x_{T_x}\}$。编码阶段本文同样引入 BiLSTM，首先将 K 编码成隐藏状态的向量$[r_1:r_{T_k}]$，将 X 编码成$[h_1:h_{T_x}]$；然后将$[r_1:r_{T_k}]$整合成一个向量 r_c，整合方法是将 BiLSTM 中前向传播的最后一个状态和反向传播第一个状态进行连接，如式(2)所示。

$$r_c=\begin{bmatrix} r_{T_k} \\ r_1 \end{bmatrix} \tag{2}$$

向量$H=[h_0:h_{T_x}]$表示关键词 K 和已生成的文本 X，其中 $h_0=r_c$，$[h_1:h_{T_x}]$表示已生成的文本。在图 3，生成第一行诗的时候，没有已生成的文本，此时，$T_x=0$，$H=[h_0]$，所以第一行诗句仅仅根据大纲的第一个关键词来生成。

在解码阶段本文引入另一个 LSTM，在 t 时刻，根据 S_t、文本语义向量 c_t 和先前的输出 y_{t-1} 生成最可能的输出 y_t，如式(3)所示。

$$y_t=\arg\max_y P(y\mid y_{t-1},c_t,s_t) \tag{3}$$

在每一时刻，S_t 按照式(4)进行更新。

$$s_t=f(s_{t-1},y_{t-1},c_t) \tag{4}$$

其中：$f(.)$是激活函数；c_t 由所有输入序列的隐层状态得出，按照式(5)计算。

$$c_t=\sum_{j=0}^{T_x}a_{tj}h_j \tag{5}$$

其中：h_j 是输入序列中第 j 个字符的编码向量；a_{tj} 为 h_j 的注意力权重，a_{tj} 被式(6)计算得出。

$$a_{tj}=\frac{\exp(e_{tj})}{\sum_{k=0}^{T_x}\exp(e_{tk})} \tag{6}$$

其中：k 从 0 开始，e_{tk} 公式如式（7）所示。

$$e_{tk}=v_a^T\tanh(W_a s_{t-1}+U_a h_j) \tag{7}$$

其中：v_a、W 和 U 是三个参数矩阵，在模型训练中需要去优化。

3　实验结果

3.1　数据处理

实验中，本文实现了格律诗的生成，格律诗有 4 行诗句，每行诗有 5 个或者 7 个汉字。本文从网上爬取了 76 475 首格律诗，从中随机挑选了 2 000 首诗作为验证集，2 000 首诗作为测试集，剩余的 72 475 首诗作为训练集。

首先，对所有古诗进行分词处理；然后计算每个词的 TextRank 分，将 TextRank 分最高的词作为每句诗的关键词，一首诗中提取出四个关键词，形式如表 2 所示。从所有训练集中共提取了 289 900 个关键词。将每首诗的关键词处理成表 3 形式用于训练 2.2.2 节中 keseq2seq 模型；本文将每首诗的诗句和对应的关键词处理成表 4 形式用于训练 2.3 节中的 pgseq2seq 模型。

表 2　绝句对应的关键词

Quatrain	Keyword
长江悲已滞	长江
万里念将归	万里
况属高风晚	高风
山山黄叶飞	山

表 3　keseq2seq 模型训练数据

Input	Output
长江	万里
长江；万里	高风
长江；万里；高风	山

表 4　pgseq2seq 模型训练数据

Input		Output
长江	——	长江悲已滞
万里	长江悲已滞	万里念将归
高风	长江悲已滞；万里念将归	况属高风晚
山	长江悲已滞；万里念将归；况属高风晚；	山山黄叶飞

3.2　模型训练

本文对 2.2.2 节中用于词扩展的 keseq2seq 模型和 2.3 节中的用于古诗生成的 pgseq2seq 模型分别进行了训练，两个模型不同是 pgseq2seq 模型有两个编码器，但是编码器的构造都是相同的，两个模型训练的方法都参考 Wang 等人[13]的序列到序列模型训练方法。模型训练目标都是让预测序列和原序列相同，本文将预测的数据分布与真实数据分布的交叉熵作为训练的损失函数，优化器采用小批量随机梯度下降算法（the minibatch stochastic gradient descent algorithm）。另外，采用 AdaDelta 算法去调节学习率[25]。最后，根据在验证集上的困惑度来选取模型最优参数。

3.3　评估方法

评估一首古诗的好坏，需要从多个维度去判定，并要求评估者具备一定的专业知识，所以评估古诗具有很大难度。目前还没有一种专门的自动评估方法用于古诗生成评估，尽管在文献[13,26]中都采用了 BLEU 的自动评估方法，BLEU 评估方法和人工评估有一定的相关性，但是 BLEU 和人工评估相比还不足以完全体现出一种古诗生成方法的有效性。就目前来看，人工评估在古诗生成任务上是一种最有效的评估方法。在本文中采用人工评估方法对比本文方法和基准方法。参考文献[10～12]的评估思路，从“前后押韵、语言流畅、内容一致、意义”四个部分去判断生成的古诗好坏，每个部分设置最高分为 5 分，得分越高越好。让每种方法对应的古诗生成系统分别生成 20 首五言绝句和 20 首七言绝句，然后邀请 20 位都具有硕士学历及以上的学者对所有生成的绝句分别打分，最后取四个部分得分的平均作为最后得分。

3.4　实验结果分析

本文中对比了四种基准方法，并对所有的方法都作了相同处理，四种基准方法如下：SMT[11]、RNNLM[27]、RNNPG[12]、ANMT[23]。表 5 中展示了人工评估的结果，在图 4 和 5 中以柱状图的形式同样展示了评估结果。

从结果中可以看出，本文所提方法在五言和七言绝句生成中都优于所有基准方法。结果显示 SMT 方法在前后押韵上优于 RNNLM 方法，这说明了基于翻译原理的方法比语言模型生成方法更能学习到前后诗句的押韵关系；ANMT 方法比 SMT、RNNLM、RNNPG 方法都表现得优越，但是劣于本文方法；本文方法和 ANMT 都采用了基于注意力机制的序列到序列的生成模型，不同的是本文是根据事先构建的大纲去生成每一句诗。

从 ANMT 和本文方法对比来看，在前后押韵和语言流畅上面本文方法提高不是很多，但是在内容一致和意义上得到了很大的提高。这正是得益于方法中大纲的构建，根据大纲中关键词和先前的诗句作为具有双编码器和注意力机制的序列到序列古诗生成模型的输入，生成的诗会让整首诗所表达的内容更加一致。另外，大纲中关键词之间的联系大大提高了诗所表现出来的意义，也让诗所表现出来的主题情感更加明确，所以，最终本文的方法在平均分上远高于 ANMT 以及所有的基准方法。

表 5　人工评估得分对比

Approachs	Poeticness		Fluency		Coherence		Meaning		Average	
	5-char	7-char	5-char	7-char	5-char	7-char	5-char	7-char	5-char	7-char
SMT	3.18	3.16	2.75	2.79	2.52	2.56	2.72	2.54	2.79	2.76
RNNLM	2.62	2.57	3.04	3.26	2.98	2.74	2.84	2.96	2.87	2.88
RNNPG	3.73	3.49	3.59	3.38	3.04	3.19	3.19	2.84	3.38	3.23
ANMT	**4.51**	4.35	4.34	4.28	3.77	3.86	3.78	3.85	4.01	4.02
Our approach	4.43	**4.39**	**4.49**	**4.54**	**4.27**	**4.39**	**4.16**	**4.36**	**4.37**	**4.42**

3.5　生成示例

表 6 列举了实验中人工交互生成的两首古诗。根据用户输入“清明怀古”和“看明月，思故乡”生成古诗。首先，从用户输入文本中提取出关键词 “清明；怀古”和“明月；故乡”；然后，对提取出的关键词进行扩展，组成大纲“清明；怀古；

酒；萧然”和“明月；故乡；惆怅；感伤”，显而易见的是提取出的关键词与扩展出的关键词之间紧密联系，关键词组合紧扣写作意图且能够表达出一种情感；最后，根据大纲的关键词生成每一行诗句，从生成的古诗中很容易看出第一首诗表现出的是怀古伤感之情，第二首诗表现的是思乡惆怅之情。

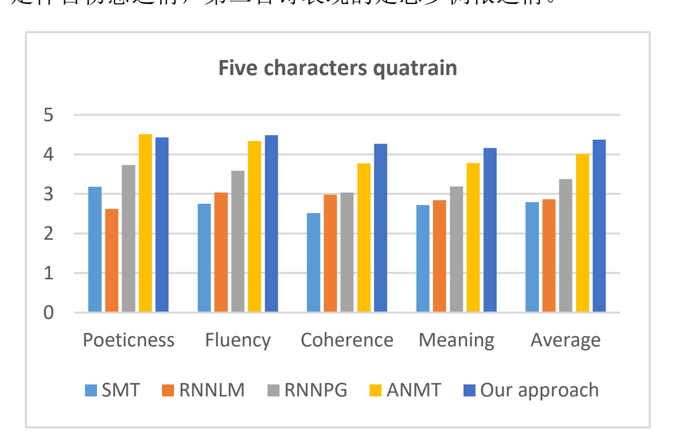


图 4　五言绝句得分统计图

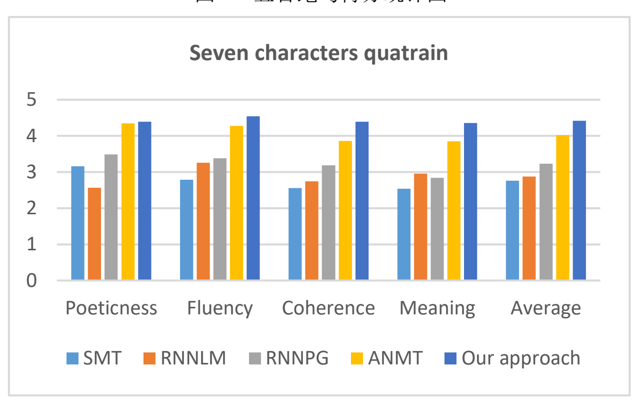


图 5　七言绝句得分统计图

表 6　古诗生成示例

Input	清明怀古	看明月，思故乡
Outline	清明；怀古；酒；萧然	明月；故乡；惆怅；感伤
Poetry	清明堂上草初长， 怀古风流晋宋王。 尽把风前一杯酒， 萧然三日菊花黄。	明月满天秋水清， 故乡风雨岁时情。 一尊惆怅人间月， 江水无情只感伤。

4　结束语

本文提出了一种将古诗生成分成两个阶段的方法，第一阶段根据用户输入文本提取作诗的大纲，第二阶段根据大纲利用具有双编码器的基于注意力机制的序列到序列模型顺序地生成每一行诗句。在第一阶段中，大纲由 N 个具有联系的关键词组成，关键词的获取首先从用户输入文本中提取，然后提出一种基于注意力机制的序列到序列词扩展模型对提取的关键词进行扩展。在第二阶段中，根据 N 个关键词来生成具有 N 行诗句的诗，每一个关键词作为每一行诗句的概要。第二阶段中，本文对基于注意力机制的序列到序列模型进行改进，改进后模型具有双编码器，然后将关键词和已生成的诗句作为两个编码器的输入，顺序地生成每一行诗句。在实验阶段，采用人工评估的方法，邀请了 20 位具有硕士学位以上的学者对本文方法以及基准方法进行了打分，最终的得分证明了本文方法优于所有基准方法。从评估结果看，本研究取得了很好的成果，对古诗生成及其他自然语言生成的研究将会有很大的参考价值。未来工作中，将在第一阶段的大纲提取中加入主题模型，如采用 PLSA、LDA 等主题模型。另外，将尝试把本文方法应用于其他自然语言生成任务上。

参考文献：

[1] Li Wang. A summary of rhyming constraints of Chinese poems (Shi Ci Ge Lu Gai Yao) [M]. Beijing: Beijing Press, 2002.

[2] Naoko T, Hideto O, Michihiko M. Hitch haiku: an interactive supporting system for composing haiku poem [C]// Proc of Entertainment Computing. Berlin: Springer, 2008: 209-216.

[3] Wu Xiaofeng, Naoko T, Ryohei N. New hitch haiku: an interactive renku poem composition supporting tool applied for sightseeing navigation system [C]// Proc of Entertainment Computing. Berlin: Springer, 2009: 191-196.

[4] Yael N, David G, Goldberg Y, *et al.* Gaiku: generating haiku with word associations norms [C]// Proc of Workshop on Computational Approaches to Linguistic Creativity. Stroudsburg, PA: Association for Computational Linguistics, 2009: 32-39.

[5] Oliveira H G. Automatic generation of poetry: an overview [D]. [S. l.] : Universidade de Coimbra, 2009.

[6] Oliveira H G. PoeTryMe: a versatile platform for poetry generation [C]// Proc of ECAI Workshop on Computational Creativity, Concept Invention, and General Intelligence. 2012: 21.

[7] Ruli M, Graeme R, Henry T. Using genetic algorithms to create meaningful poetic text [J]. Journal of Experimental & Theoretical Artificial Intelligence, 2012, 24 (1): 43–64.

[8] Manurung H M. An evolutionary algorithm approach to poetry generation [D]. [S. l.] : University of Edinburgh, 2004.

[9] Zhou Chengle, You Wei, Ding Xiaojun. Genetic algorithm and its implementation of automatic generation of chinese songci [J]. Journal of Software, 2010, 21 (3): 427–437.

[10] Yan Rui, Jiang Han, Lapata M, *et al.* I, poet: automatic chinese poetry composition through a generative summarization framework under constrained optimization [C]// Proc of the 23rd International Joint Conference on Artificial Intelligence. 2013.

[11] He Jing, Zhou Ming, Jiang Long. Generating Chinese classical poems with statistical machine translation models [C]// Proc of the 26th AAAI Conference on Artificial Intelligence. 2012.

[12] Zhang Xingxing, Mirella L. Chinese poetry generation with recurrent neural networks [C]// Proc of Conference on Empirical Methods in Natural Language Processing. [S. l.] : Association for Computational Linguistics, 2014: 670-680.

[13] Wang Qixin, Luo Tianyi, Wang Dong, *et al.* Chinese song iambics generation with neural attention-based model [EB/OL]. (2016-06-21) . https://arxiv. org/abs/1604. 06274.

[14] Wang Zhe, He Wei, Wu Hua, *et al.* Chinese poetry generation with planning based neural network [C]// Proc of the 26th International Conference on Computational Linguistics. 2016: 1051-1060.

[15] Yi Xiaoyuan, Li Ruoyu, Sun Maosong. Generating Chinese classical poems with RNN encoder-decoder [EB/OL]. (2016-04-06) . https://arxiv. org/abs/1604. 01537.

[16] Greene E, Bodrumlu T, Knight K. Automatic analysis of rhythmic poetry with applications to generation and translation [C]// Proc of Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: Association for Computational Linguistics, 2010: 524-533.

[17] Simon C, Jacob G, Tony V. Full-face poetry generation [C]// Proc of ICCC. 2012.

[18] Jiang Long, Zhou Ming. Generating chinese couplets using a statistical mt approach [C]// Proc of the 22nd International Conference on Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2008: 377-384.

[19] Mikolov T, Karafiát M, Burget L, *et al.* Recurrent neural network based language model [C]// Proc of the 11th Annual Conference of the International Speech Communication Association. 2010: 3.

[20] Ghazvininejad M, Shi Xing, Choi Y, *et al.* Generating topical poetry [C]// Proc of Conference on Empirical Methods in Natural Language Processing. 2016.

[21] Ilya S, Oriol V, Le Q V. Sequence to sequence learning with neural networks [C]// Advances in Neural Information Processing Systems. 2014: 3104-3112.

[22] Rada M, Paul T. TexTrank: bringing order into text [C]// Proc of EMNLP. 2004.

[23] Brin S, Page L. The anatomy of a large-scale hypertextual Web search engine [J]. Computer Networks, 1998, 30: 107-117.

[24] Dzmitry B, Kyunghyun C, Yoshua B. Neural machine translation by jointly learning to align and translate [J]. arXiv preprint arXiv: 1409. 0473, 2014.

[25] Zeiler M D. Adadelta: an adaptive learning rate method [EB/OL]. (2012-12-22) . https://arxiv. org/abs/1212. 5701.

[26] 蒋锐滢，崔磊，何晶，等. 基于主题模型和统计机器翻译方法的中文格律诗自动生成 [J]. 计算机学报, 2015, 38 (12): 2426-2436. (Jiang Ruiying, Cui lei, He Jing, *et al.* Automatic generation of Chinese metrical poems based on thematic models and statistical machine translation methods [J]. Journal of Computer Science, 2015, 38 (12): 2426-2436.)

[27] Graves A. Generating sequences with recurrent neural networks [EB/OL]. (2014-06-05). https://arxiv.org/abs/1308.0850.